Computers and Chemical Engineering 106 (2017) 373–384

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

A decomposition framework for managing inventory and distribution of fluid products by an order-based-resupply methodology

Mariana E. Cóccola, Carlos A. Méndez, Rodolfo G. Dondo*

INTEC (Universidad Nacional del Litoral - CONICET), Güemes 3450, 3000 Santa Fe, Argentina

ARTICLE INFO

Article history:
Received 25 October 2016
Received in revised form 15 June 2017
Accepted 19 June 2017
Available online 28 June 2017

Keywords:
Distribution of fluids
Order-based-resupply
Incomplete branch-and-price
Inventory constraints

ABSTRACT

Fluid chemicals are usually distributed according two main methodologies; the vendor managed inventory modality and the order-based re-supply modality. In this paper, the problem of optimizing the delivery of fluids by tanker trucks on a daily basis according the order-basedresupply methodology is addressed. According to this modality, replenishment orders are triggered by customers specifying the quantity of fluids and time-windows within which the delivery must be fulfilled. The objective is to minimize the replenishment cost while meeting customers orders over the pre-defined time-horizon. An Integer Program modelling the problem is proposed and used to develop an incomplete branch & price procedure with the purpose of finding near optimal solutions to instances arising from realistic examples. A computational study on realistic examples with different topologies demonstrates that the method is effective and able to provide solutions with integrality gaps below the 16% threshold for instances with 120 orders.

1. Introduction

The scale of the chemical industry is global and logistics is a crucial part of this industry because raw materials sources, production facilities and consumer markets are globally distributed. Due to the increasing pressure for reducing costs, inventories and ecological footprint, and in order to remain competitive in the global marketplace, Enterprise-wide Optimization (EWO) has become a major goal of the chemical industry (Grossmann, 2012). In this way, fluctuating demand, seasonal imbalances of raw materials flows and final products flows, as well as expensive transportation motivate a dynamic and integrated management of logistic activities.

The transportation of chemical liquids and gases involves several operations and each one represents a potential source of savings. For example, crude oil is transported from production fields to refineries by tankers, barges, rail-tank cars, tank-trucks, and pipelines. Refined petroleum products are conveyed to fuel terminals and petrochemical plants on similar ways. From fuel-hubs and petrochemical plants, fuels and chemical fluids are delivered by tank trucks to service fuel stations, commercial accounts, and local bulk storage plants. The final destination for gasoline is usually a motor vehicle gasoline tank but fairly similar distribution channels exist for other fluid chemical products. A general depiction of these distribution channels is shown in Fig. 1.

Customers of industrial gases or/and liquids are usually serviced by pipeline and bulk truck delivery, respectively. Customers of gaseous products are usually located near the plants and referred as "on-site" or "over-the-fence" customers Their demands are tied by contractual obligations that must always be met (Marchetti et al., 2014). Liquid products are usually stored in on-site storage-tanks and from there they are usually loaded into trailers and carried to customer sites by trucks. The transportation cost for bulk truck delivery is the main component of the distribution cost and inventory costs are substantially lower (Marchetti et al., 2014). Both the frequency of deliveries to a given customer and the selection of routes supplying multiple product to customers are critical to reduce transportation costs. The distribution schedule depends not only on the availability of tractors, trailers, and drivers, but also on the inventory levels of fluid products at plants/hubs and at customers. In the distribution of perishable products, the supply to various locations in a particular geographic or administrative region is usually met by one or more regional suppliers. In both cases, the separate treatments of allocation of supply-sources and distribution can result in a poor performance of the system as a whole. So, suppliers must determine when, from where and how much to deliver to customers over a finite planning horizon. Replenishment activities may be triggered either by a customer order or may be determined by the vendor concurrently with the

* Corresponding author.
E-mail address: rdondo@santafe-conicet.gov.ar (R.G. Dondo).

http://dx.doi.org/10.1016/j.compchemeng.2017.06.022

Nomenclature

Sets

C	Trailer compartments
I^+	Plants/distribution hubs
I^-	Customers
P	Products
R	Feasible routes

Parameters

α_{ipr}	Binary parameter denoting that order of product p by customer i belongs to route r
a_i	Earliest arrival time at customer i
b_i	Latest arrival time at customer i
$cv_{tractor}$	Variable cost of the tractor used in the designed route
$cv_{trailer}$	Variable cost of the trailer used in the designed route
c_r	Cost of route r
c^t	Overcost due to a unit-time violation of a time windows
d_{ij}	Distance between locations i and j
$fc_{tractor}$	Fixed cost of the tractor used in the designed route
$fc_{trailer}$	Fixed cost of the trailer used in the designed route
ft_i	Fixed stop time at location i
l_{ip}	Stock of product p on plant/hub i
l_r	(Un)loading rate
$q_{tractor}$	Tractor capacity
$q^c{}_{trailer}$	Capacity of compartment c of the trailer
st_i	Total service time at location i
t_{ij}	Travel time between locations i and j
α_{ipr}	Quantity of product p loaded from site i to deliver by route r
β_{ip}	Demand of product p by customer i
$\pi_{ip}{}^+$	Price associated to the inventory-constraint of product p on plant/hub i
$\pi_{ip}{}^-$	Price associated to the demand of product p by customer i

Binary variables

S_{ij}	Variable for sequencing locations i and j along a route
X_r	Variable for selecting route r
Y_i	Variable used to determine that the site i belongs to a route designed by a slave routes-generator problem
$T^{trailer}$	Variable indicating that a trailer is used by a vehicle
$V_p{}^{tractor}$	Variable to allocate the product p to the tractor used in the designed route
$V_{pc}{}^{trailer}$	Variable to allocate the product p to the compartment c of the trailer used in the designed route
Z_{ip}	Variable used to determine that the order of product p by customer i belongs to the route designed by a slave routes-generator problem

Continuous variables

CV	Cost of the route designed by a slave routes-generator problem
$CV_{tractor}$	Cost of the tractor used on the designed route
$CV_{trailer}$	Cost of the trailer used on the designed route
D_i	Distance travelled to reach location i
T_i	Time spent to reach location i
TV	Time spent on the route designed by the slave routes-generator problem

quantity of products to deliver within the so called "regeneration cycle". Replenishment triggered by a customer or "order-based-resupply" (OBR) corresponds to customers which send orders to the vendor, specifying the desired quantity and the time window in which the delivery must be done (Benoist et al., 2009) while in the "vendor-managed-inventory" (VMI) or "forecasting based resupply" modality, each product-inventory of each customer must be replenished by tank trucks so as to never fall under the safety level. Constraints consisting in maintaining inventory levels above safety levels (no stock out) and below maximum capacity are usually considered soft constraints. Also, some customers may ask for the both types of resupply modalities to keep the possibility of ordering to deal with an unexpected increase of the consumption rate (Benoist et al., 2009).

Land transportation of fluid products is usually performed by triplets composed of three resources; drivers, tractors, and trailers. A "vehicle" corresponds to the association of one driver, one tractor and one trailer. Some triplets of resources are not feasible and each plant or customer is accessible just by a subset of vehicles because special skills or certifications are required to work on certain sites (Van der Bruggen et al., 1995). The planning of the resupply schedule consists on designing a set of "shifts" or vehicle-tours comprising operations of pickups on plants and of deliveries to customers. The so called shift starts from the warehouse to which the resources composing a vehicle are allocated and usually must end by returning to this warehouse. The working and driving times of drivers are limited and as soon as a maximum duration is reached, the driver must take a rest with a minimum duration (Ver der Bruggen, 1995). Also, customers visited along the tour should be accessible within their time-windows. Finally, for each plant/replenishment-hub, inventory constraints must be satisfied.

In the VMI case, for each product, the sum of quantities delivered to each customer minus the sum of the quantities consumed over a time-period must be lower than the its storage-capacity and larger than zero. Similar constraints are also taken into account for the quantities of products produced in any plant. For the OBR modality, storage and safety constraints are not explicitly considered on customers and just the order size is taken into account for planning replenishments. In this case, the sum of quantities of a given product delivered from a given plant must be lower then the plant inventory and although different vehicles may satisfy different product-orders to the same customer, the order of a single product is usually satisfied by a single vehicle.

This work aims at the optimal integration of production, inventorying and distribution of fluid chemicals. Its main contributions are the following:

1. The formulation of a set-partitioning model for the optimal planning of the OBR of chemical fluids. Multi-period VMI strategies are left to the next work.
2. The development of an incomplete branch-and-price algorithm for solving the formulation of the OBR methodology.
3. Computational experiments on instances featuring different characteristics are performed to test the capability of the algorithm for providing effective and efficient solutions.

The remainder of this paper is organized as follows: The literature related to the subject is reviewed in Section 2. Section 3 describes the OBR-based methodology for delivering multiple fluid products. The formulation developed to model and optimize this resupply modality is explained in Section 4. The incomplete branch and price algorithm is also detailed in this section. Numerical experiments over a series of realistic instances featuring different cases and characteristics are presented in Section 5. The concluding remarks follow in Section 6.

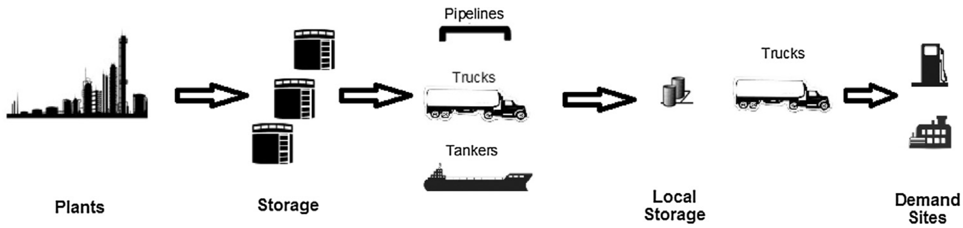

Fig. 1. Distribution of fluid chemical products from refineries/plants to demand sites.

2. Literature review

Since the classical work of Bell et al. (1983) on a real-life inventory routing problem encountered on *Air Products*, several papers researching the distribution of industrial gases emerged in the literature. A series of papers published by Savelsbergh and co-workers (Campbell et al., 1998; Campbell et al., 2002; Campbell and Savelsbergh, 2004) describe with a high degree of detail the real-life problematic encountered at *Praxair*. The solution approaches proposed in these papers involve two phases. In the first phase, it is decided which customers are visited during the following days and the amount of product to deliver to customers. In the second phase, vehicle routes are determined taking into account vehicle capacities, customer time-windows and drivers restrictions.

The benefits of integration of routing and inventory was already stated and surveyed by Baita et al. (1998) almost twenty years ago. Christiansen and co-workers published numerous works about the integration of routing and inventory management within the maritime context. They usually employ different column generation procedures and enforce inventory constraints at the master level of the procedure. We just refer to the review by Andersson et al. (2010) to see a brief introduction to this algorithmic framework as well as the proposed variants and applications. More recent works on this research path are Stålhane et al. (2012); Ghiami et al. (2015) and Hemmati et al. (2016).

The inventory routing problem (IRP) constitutes the core of VMI methodologies while the OBR modality is based on the vehicle routing problem (VRP) with additional inventory constraints for delivering products. On the other hand the OBR methodology deals with daily distribution of one or more than one product like for example perishables. Therefore, its cornerstone problem is the widely researched VRP. We just refer to the Golden et al. (2010) book for reviewing research about this classic OR problem. As the VRP does not capture all issues related to distribution of industrial commodities, Dondo et al. (2009) introduced the so called vehicle routing problem in supply chain management (VRP-SCM). The approach provides a set of optimal vehicle routes to meet demands of multiple products at minimum transportation cost. In order to efficiently address larger problems this type, a column generation procedure was proposed by Dondo and Mendez (2014). In contrast to traditional columns generation approaches lying on dynamic-programming-procedures, an MILP formulation was proposed to create the set of feasible routes and schedules at the slave level of the method. The subject is receiving a growing attention from the chemical-engineering research perspective. For example You et al. (2011) addressed the optimization of industrial gases distribution systems, which consist of plants and customers, as well as storage tanks, trucks and trailers. The locations of the plant and customers, as well as the distances between them are given. Each customer has a deterministic and constant demand rate and safety stock level. A set of possible tanks with different discrete sizes is given. For customers with existing tanks, their specific initial inventory levels are given. New customers need to determine the size of tank to be installed and existing customers can upgrade or downgrade the existing tanks, or add a second tank if extra space is available. There is a number of trucks with discrete capacities and the delivery cost per distance travelled for every truck is given. Singh et al. (2015) considered a multi-period IRP with multiple products assuming deterministic demand-rates and the proposed model is formulated as a linear mixed-integer program. The authors proposed an incremental approach based on decomposing the set of customers of the original problem into sub-problems. A sub-problem was incrementally solved with the number of customers growing successively by providing the solution of the previously solved sub-problem as an input by using a randomized local-search heuristic method. Marchetti et al. (2014) proposed a multi-period mixed-integer linear programming model for the optimal enterprise-level planning of industrial gas operations. Their objective was to minimize the total cost of production and distribution of liquid products by coordinating production decisions at multiple plants and distribution decisions at multiple depots. The proposed methodology includes an MILP model for planning the production of liquids and a heuristic procedure to design distribution routes. The main goal of the paper was to assess the benefits of optimal coordination of production and distribution. Since the computational expenses of such problems becomes a major limitation when dealing with industrial size instances, Zamarripa et al. (2016) proposed a rolling horizon methodology with two aggregation strategies for solving smaller subproblems. The first one relies on the linear programming (LP) relaxation for which the binary variables of the distribution problem are treated as continuous, while a second one uses a model tailored for the distribution side-constraints.

From an algorithmic point of view, many variants of routing and inventorying problems are currently solved by an effective decomposition technique called column generation. We just refer to the tutorial paper by Lübbecke and Desrosiers (2005) as well as to the book by Desaulniers et al. (2005) for reviewing column generation and its applications in vehicle routing, routing and inventory management as well as in another research fields.

3. Description of the OBR methodology

Distribution strategies based on the OBR methodology deal with daily distribution of several products like perishables or chemicals. To describe this modality, let us consider several customers spread over a geographical area. Each customer consumes fluid products that are moved from plants producing them. Each customer is equipped with a multi-commodity storage and similarly, each plant has a multi-commodity storage from which several products can be pumped out. Customer demands and products inventories over the short-term planning horizon are known data. For transporting several fluid products from plants to customers, the following issues must be addressed by the planner:

1. Which clients to deliver from which plant?
2. How many tractor-trailer combinations (vehicles) must be used?
3. How to assign products to compartments of each vehicle?

The solution procedure developed to deal with the OBR modality also can be used as a sub-problem in two-phases solution-

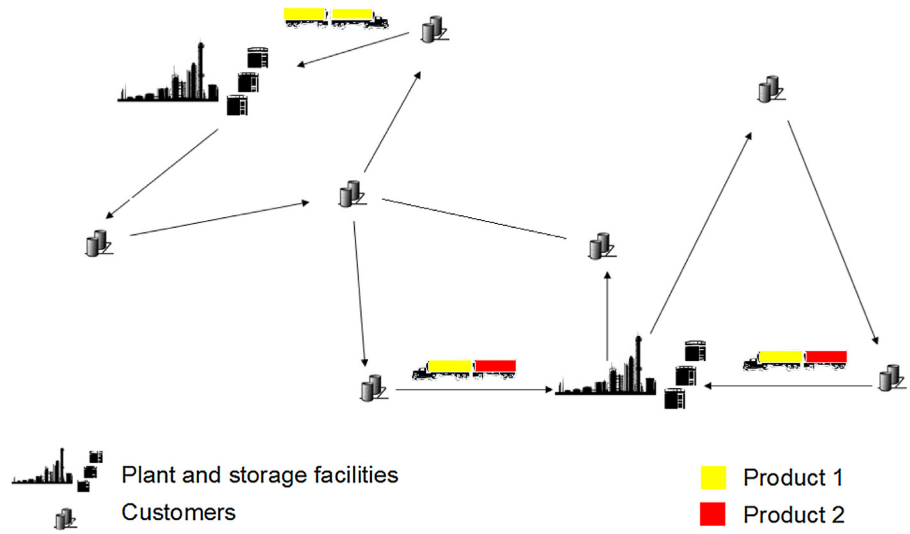

Fig. 2. Illustrating the OBR modality.

methodologies for solving VMI-like problems. In this way the sub-problem is used after fixing the delivering day and the quantity of products to deliver to each customer. The Fig. 2 illustrates the researched problematic.

Considering the complexity and the dimension of the problems, which typically involve dozens of clients and several products, in practice it is infeasible to optimally solve a monolithic model for finding solutions useful as answers to the above stated questions. Therefore, we have chosen to develop a decomposition approach based on the column generation paradigm in order to find near optimal solutions and an estimation of the sub-optimality gap of such solutions. In such an approach, the original problem is decomposed into a master-slave couple that is iteratively solved. The master problem is basically the relaxation of the integer problem used for selecting a number of routes visiting a given subset of clients to satisfy their demands subject to inventory constraints on the plants. The sub-problem defines (i) the "vehicle" as a tractor-trailer combination; (ii) the assignment of products to the compartments of the vehicle, and (iii) the routes and schedules for pick-up and delivery activities. In other words, the column generation procedure provides the following information:

1. The number of vehicles departing from each plant/hub.
2. The tractor-trailer combination defining each vehicle.
3. The quantity and type of products transported by each compartment of each vehicle.
4. The shifts and their detailed delivery schedules.

The costs involved in the optimization of the problem are the following:

1. Fixed transportation costs for using a tractor ($cf_{tractor}$) and using a trailer ($cf_{trailer}$).
2. Variable transportation costs for using a tractor ($cv_{tractor}$) and using a trailer ($cv_{trailer}$).
3. Over-costs due to missing customers time windows (c_t).
4. Inventory costs in plants are considered much lower than transportation costs and therefore are neglected.

The parameters that must be taken into account to model and solve the problem are:

1. The order size per product $p \in P$ by any client $i \in I^-$ (β_{ip}).
2. The inventory of product $p \in P$ on any plant $i \in I^+$ (l_{ip}).
3. Tractor ($q_{tractor}$) and trailer-compartment ($q^c{}_{trailer}$) capacities.
4. Fixed (ft_i) and variable times (l_t) required to load a tractor-trailer combination on a plant $i \in I^+$ or to unload on a customer $i \in I^-$.
5. Estimated travelling times (t_{ij}) and distances (d_{ij}) between a pair of clients, and between the plants and the clients.
6. Hours of the day within which clients $i \in I^+$ can be serviced or "time windows" (a_i, b_i).

4. Model formulation and solution methodology

In order to formulate the above described distribution-problematic as an Integer Program (IP), let us denote R as the set of all feasible pickup and delivery routes. For each route $r \in R$, c_r denotes its cost, which is determined by the sum of the costs of the arcs travelled by the vehicle plus the fixed tractor and trailer utilization costs plus eventual overtime costs. Let us consider a binary parameters a_{ipr} indicating whether route $r \in R$ fulfills ($a_{ipr} = 1$) or not ($a_{ipr} = 0$) the demand of product $p \in P$ by the client $i \in I^-$. For a route $r \in R$, let us consider also a positive parameter α_{ipr} that takes a value equal to the quantity of product $p \in P$ loaded on plant $i \in I^+$ and transported on the vehicle fulfilling all demands allocated to the route $r \in R$. To determine if the route $r \in R$ belongs to the optimal solution we use the binary decision variable X_r, and then, the problem can be stated as the following IP:

Minimze

$$\sum_{r \in R} c_r X_r \tag{1}$$

subject to:

$$\sum_{r \in R} a_{ipr} X_r = 1 \quad \forall i \in I^-, p \in P \tag{2}$$

$$\sum_{r \in R} \alpha_{ipr} x_r \le l_{ip} \qquad \forall i \in I^+, p \in P \tag{3}$$

$$X_r \in \{0,1\}$$

The formulation considers, in addition to partitioning constraints (2), the inventory constraints (3) for each product $p \in P$ on each plant $i \in I^+$. We remark that constraints (2), unlike common reformulations of routing problems that generally use covering constraints, are partitioning constraints. That's because, for the delivery of multiple items to a single customer via one or several vehicles, a partitioning solution equivalent to the optimal covering solution may not exist (Salani and Vacca, 2009). This constraint states that the demand of each product by each customer must be satisfied exactly once. Constraint (3) states that the total quantity of product $p \in P$ delivered from plant $i \in I^+$ must not exceed the available product inventory on the plant.

Usually, realistic instances are large and consequently all feasible tours cannot be enumerated. Therefore, column generation approaches have been used to solve path-based models like this one defined by Eqs. (1)–(3). Early research on column generation was performed by Appelgren (1969) and during the 1990s the methodology was successfully extended to numerous variants of the VRP. Nowadays, an extraordinary number of applications can be found in the literature (See for instance Lübbecke and Desrosiers, 2005).

The column generation approach implicitly considerers all feasible tours through the solution of the linear relaxation of model (1)-(3). In this way, a feasible initial solution defined by a few shifts (the so called reduced master problem, RMP) is enumerated and the linear relaxation of the formulation (1)-(3) is solved considering just this set. The solution to this problem is used to determine if there are routes not included in the pool of available routes that can reduce its objective function value. Using the value of the optimal dual variables with respect to the partial set, new shifts are generated and the linear relaxation is solved again. The procedure iterates until no tours with negative reduced costs can be found. Finally, the integer formulation of the model (1)-(3) may be solved for finding the best subset of routes but the solution found may not be the global optimal. To find the optimal one, the procedure must be embedded into a branch-and-bound algorithm because some routes that were not generated when solving the relaxed RMP may be needed to find the optimal integer solution. The whole process is named branch-and-price, which involves the definition of the RMP, the definition of the slave or pricing problem and the implementation of a branching rule. All of them are detailed in the following subsections.

4.1. The reduced master problem

Decomposition procedures based on column generation use route selection variables in the master program. Each route has an associated binary variable X_r. Dual-variable-values can be obtained by enumerating a feasible solution comprising a partial set of shifts $R' = \{r_1, r_2, \ldots r_{r'}\}$ and by solving the linear relaxation of the RMP. So, let us assume that c_r is the cost of route r and let $\boldsymbol{\pi_I} = \{\pi^-{}_{i1,p1}, \ldots\}$ and $\boldsymbol{\pi_c} = \{\pi^+{}_{i1,p1}, \ldots\}$ the vectors of optimal dual variables values associated to constraints (2) and (3) respectively. When the slave problem is solved and the reduced cost of a new route is negative, a route potentially improving the overall LP solution has been discovered. Then, c_r $\boldsymbol{\pi_I}$ and $\boldsymbol{\pi_c}$ are optimal for the current RMP but not optimal for the global master problem. So, the shift just found is added to the partial shifts-set and a new RMP must be solved again until no shifts with negative reduced costs can be found.

4.2. The slave problem

Each shift or vehicle-tour can be represented as an elementary path that includes some customers and starts and ends at the same warehouse. Usually the warehouse is located near a production-plant but this is not always the case. The slave or pricing problem is an elementary shortest path problem with resource constraints (ESPPRC) and it is NP-hard in the strong sense. There are several techniques to solve the pricing problems as dynamic programming, constraints programming or MILP programming. In the current application, we solve the MILP formulation of the ESPPRC with a branch-and-cut solver generating several shifts with negative reduced costs per iteration. The objective of the slave problem is to find a shift that minimizes the quantity stated by Eq. (4) while satisfying the constraints stated by Eqs. (5)–(12):

Minimize

$$\left[CV + c^t\left(\Delta a_i + \Delta b_i\right) - \sum_{i \in I^-}\sum_{p \in P}\pi^-_{ip}Z_{ip} - \sum_{i \in I^+}\sum_{p \in P}\pi^+_{ip}L_{ip} \right] \tag{4}$$

Subject to

$$\sum_{i \in I^+} Y_i = 1 \quad \forall \in I^+ \tag{5}$$

$$D_i \geq d_{ji} Y_j \qquad \forall i \in I^+, j \in I^- \tag{6.a}$$

$$\begin{Bmatrix} D_j \geq D_i + d_{ij} - M_D(1-S_{ij}) - M_D(2-Y_i-Y_j) \\ D_i \geq D_j + d_{ji} - M_D S_{ij} - M_D(2-Y_i-Y_j) \end{Bmatrix} \qquad \forall (i,j) \in I^- : i < j \tag{6.b, 6.c}$$

$$CV_{tractor} \geq \left(cf_{truck} \sum_{p \in P} V_p^t + cv_{truck}(D_i + d_{ij}Y_j) \right) - M_C\left(1 - \sum_{p \in P} V_p^{truck}\right) \qquad \forall i \in I^+, j \in I^- \tag{6.d}$$

$$CV_{trailer} \geq \left(cf_{trail} \sum_{p \in P} V_p^t + cv_{trail}(D_i + d_{ij}Y_j) \right) - M_C\left(1 - T^{trailer}\right) \qquad \forall i \in I^+, j \in I^- \tag{6.e}$$

$$CV \geq CV_{tractor} + CV_{trailer} \tag{6.f}$$

$$\begin{Bmatrix} T_j \geq T_i + st_i + t_{ij} - M_T(1-S_{ij}) - M_T(2-Y_i-Y_j) \\ T_i \geq T_j + st_j + t_{ji} - M_T S_{ij} - M_T(2-Y_i-Y_j) \end{Bmatrix} \qquad \forall (i,j) \in I^+ \cup I^- : i < j \tag{7.a, 7.b}$$

$$TV \geq T_i - M_T(1-Y_i) \qquad i \in I^+ \cup I^- \tag{7.c}$$

$$\begin{Bmatrix} a_i - T_i \geq \Delta a_i \\ T_i - b_i \geq \Delta b_i \end{Bmatrix} \qquad i \in I^+ \cup I^- \tag{8.a, 8.b}$$

$$Z_{ip} \leq Y_i \quad \forall i \in I^-, p \in P \tag{9}$$

$$\sum_{i \in I^-} Z_{ip}\beta_{ip} \leq \mathrm{q}v_{tractor} V_p^{tractor} + \mathrm{q}v^c \ trail \sum_{c \in C} V_{pc}^{trailer} \quad \forall p \in P \tag{10}$$

$$\sum_{p \in P} V_p^{ttractor} \geq \sum_{p \in P} V_{pc}^{trailer} \quad \forall c \in C \tag{11.a}$$

$$\sum_{p \in P} V_p^{tractor} \leq 1 \tag{11.b}$$

$$\sum_{p \in P} V_{pc}^{trailer} \leq 1 \quad \forall c \in C \tag{11.c}$$

$$M_C T^{trailer} \leq \sum_{c \in C} \sum_{p \in P} V_{pc}^{trailer} \tag{12.a}$$

$$\sum_{p \in P} V_{pc}^{trailer} \geq \sum_{p \in P} V_{pc'}^{trailer} \quad \forall (c, c') \in C : c < |C|, c' = c + 1 \tag{12.b}$$

The objective function (4) is the cost *CV* of the shift plus eventual over-costs due to time-windows violations minus the collected prices associated to the supply of product $p \in P$ on each visited customer $i \in I^-$ and minus prices associated to inventory constraints of product $p \in P$ on the supplying plant $i \in I^+$. Values for these prices are obtained from the duals of constraints (2) and (3) respectively after solving the relaxed RMP. Constraints (5) select the plant from where the vehicle starts and ends its shift. The *cost related constraints* given by Eq. (6) compute the distances travelled to reach the visited sites $i \in I^-$ and the total cost of the generated route. So, Eq. (6.a) set the minimum distance to go from the selected plant to any visited customer. Eqs. (6.b) and (6.c) fix the accumulated distance up to each visited site. I.e. if nodes *i* and *j* are allocated to the generated route ($Y_i = Y_j = 1$), the visiting ordering for both sites is determined by the value of the sequencing variable S_{ij}. If location *i* is visited before *j* ($S_{ij} = 1$), according constraints (6.b), the travelled distance up to the location *j* (D_j) must be larger than D_i by at least d_{ij}. In case node *j* is visited earlier, ($S_{ij} = 0$), the reverse statement holds and constraint (6.c) becomes active. If one or both nodes are not allocated to the tour, Eqs. (6.b) and (6.c) become redundant. M_D is an upper bound for variables D_i. Eq (6.d) computes the cost $CV_{tractor}$ of the vehicle-tractor as the product of the cost coefficient c_v by the travelled distance and the addition of the fixed truck utilization cost ($cf_{tractor} \ \Sigma_p V^{tractor}$). Eq. (6.e) is similar to Eq. (6.d) but it computes the trailer cost just in case the vehicle utilizes a trailer. Eq. (6.f) determines the total vehicle cost. M_C is an upper bound for the variables $CV_{tractor}$ and $CV_{trailer}$. The *timing constraints* stated by Eq. (7) define visiting-time constraints that are similar to constraints (6.a) and (6.b) but apply to the time dimension. M_T is an upper bound for the times T_i spent to reach the visited nodes. Eq. (8) forces the service time on any site $i \in I$ to start at a time T_i bounded by the time window (a_i, b_i). Nevertheless the time windows can be violated at a cost proportional to either one of the time violations Δa_i or Δb_i. The *load related constraints* are stated by Eqs. (9)–(12). Eq. (9) states that no cargo can be taken from plant $i \in I^+$ if the vehicle doesn't visit that location. Eq. (10) sets the total cargo of product *p* loaded by the travelling vehicle by allocating the cargo to the tractor and/or to the compartment of the trailers. The vehicle capacity is composed by the truck capacity and by the total trailer capacity, given by the sum of the capacities of its compartments. Eq. (11.a) states that not a compartment of the trailer can be used if a tractor is not used. Eq. (6.b) allocates the truck capacity to one product and Eq. (11.c) allocates the capacity of the compartments of the trailer also to just one product. Note that the tractor-trailer combination as well as the mix of product to deliver are decided by the solution of the problem. Constraint (12.a) states that no a trailer-compartment can be filled if the trailer is not used. Eq. (12.b) sets the order for filling compartments. This is a symmetry-breaking constraint aimed at eliminating symmetrical solutions sharing the same objective function value. Since this formulation uses several big-M terms and presents weak linear relaxations, pre-processing rules and valid inequalities developed by Dondo (2012) were used to strength the formulation and speed-up its resolution. These are additional constraints consisting on inequalities involving binary variables that are useful for considerably speeding-up the resolution process.

4.3. Branching strategy

The linear relaxation of the RMP may not be integer and applying a standard branch-and-bound procedure to this problem with a given pool of columns may not yield an optimal solution (Barnhart et al., 2000). That's because some non-generated columns may be needed in the RMP. Consequently, if the relaxation of the RMP is not integer, to find an optimal solution, columns must be generated after branching. According to Savelsbergh and Sol (1998), it is good to utilize a branching scheme that focuses on assignment decisions rather than on routing decisions because assignment decisions constitute higher level decisions and have a greater impact on the structure of the solution. Based on the above idea we propose the following branching scheme: when the solution is not integer, the procedure finds a customer $i \in I^-$ and a plant $j \in I^+$ and creates two subspaces; on one both sites must be visited along the same shift and on the other one they must not be on the same shift. This branching scheme can be viewed as a special case of the branching rule proposed by Ryan and Foster (1981). Branching decisions are imposed as constraints and usually do not complicate the pricing problem in the subspaces. The procedure is similar to the one proposed by Savelsbergh and Sol (1998) that selects an individual vehicle and an individual customer to create two sub-spaces but in our case, the vehicle is replaced by a supply plant. When solving the pricing problem, the first restriction is satisfied by forcing locations $i \in I^-$ and $j \in I^+$ to be both visited. The second branching restriction requires a constraint stating that whenever $j \in I^+$ is on the tour, then the visit to $i \in I^-$ is not allowed or viceversa. We call the DIFFER-child the one that, by fixing all variables to zero, cover locations $i \in I^-$ and $j \in I^+$ and they will be fulfilled by two different vehicles with solution value 1. If both locations $i \in I^-$ and $j \in I^+$ have to be visited along the same tour we call this child the SAME-child. To enforce branching constraints, it is necessary just to exclude columns from the solution space defined by the columns-set considered in the branch-and-price node (by setting its associated variable bounds to zero) and to enforce them at the pricing level either by eliminating the customer(s)/plant from the graph and/or by forcing the tour to visit the fixed customer(s) and the fixed plant.

4.4. Implementation and numerical issues

The algorithm developed has been coded in GAMS 23.6.2. This procedure integrates a column generation routine (comprising the relaxed RMP detailed and the slave-problem) into a branch-and-bound procedure that branches according to the rule proposed in Section 4.3. Both GAMS routines were built over routines developed by Kalvelagen (2009, 2011). They were integrated in this work to lead to the current decomposition procedure. Some branching and assembling modifications aimed at replacing the NLP of the Kalvelagen (2009) MINLP algorithm by the column generation procedure and aimed at fixing couples (*i*,*j*): $i \in I^-$, $j \in I^+$ along the branching tree were introduced. The branching mechanism is described in the left column of Fig. 3. The right column provides a brief explanations of its row aligned sentences. The algorithm uses CPLEX 11 as the MILP sub-algorithm for generating columns and

```
Loop(node,                                                   Nodes processing loop. It selects the
     bestbound ?  min_waiting(n) bound(n);                   minimum bound waiting node.
     current(n) ?  no;
     current(waiting(n)): bound(n) = bestbound  ?  yes;
    Loop(current: first, first ?  0;                         Processing the first node current(n) of
       waiting(current) ?  no;                               the waiting list waiting(n).
   Loop(i: i ∈ I+, loop(j: j ∈ I-, , fx_ij ?  no));          Fixing the branching pair (i,j).
   Loop(i: i ∈ I+, loop(j j ∈ I-: (fx^1(current,i,j) or fx^0(current,i,j)), fx_i ?  yes);
    Loop(n: current(n) and not waiting(n),                   Selecting columns allocated to the
       Loop(i: i ∈ I+,                                       current(n) node by excluding columns
        Loop(j: j ∈ I- and fx_ij and fx^0(current,i,j),     recorded in the list exception(r).
          Loop(r, exception(r) ?  no;
           if(a_ir = 1 and a_jr = 1, then exception(r) ?  yes);
             );
           );
         Loop(j : j ∈ I- and fx_ij and fx^1(current,i,j),
          Loop(r, exception(r) ?  no;
           if(a_ir = 1 and a_jr = 0,  then exception(r) ?  yes);
           if(a_ir = 0 and a_jr = 1,  then exception(r) ?  yes);
             );
           );
         );
Run Column Generation                                        Solving the master-slave iteration
                                                             until no more columns are generated.
LLB = Obj. Function (linear RMP on the current node columns) Computes the LLB.
   Loop(p,                                                   Update duals.
      Loop(i: i ∈ I-,  π-_ip = Duals from constraint (2));
      Loop(i: i ∈ I+,  π+_ip = Duals from constraint (3));
     );
            If(LLB = integer,                                Updating the best integer solution
              if(LLB < bestfound,                            and record it in the list best(r,i).
              bestfound = LLB;
              Loop(r, bestx(r) = x_r;
                        Loop(i, best(r,i) = a_ir));
              Else
               GUB = Obj. Function (integer RMP on all columns)
              if(GUB < bestfound,
              bestfound = GUB;
              Loop(r, bestx(r) = x_r;
                        Loop(i, best(r,i) = a_ir));
              );
               wait(n) ?  no; wait(waiting) ?  yes;          Terminating waiting nodes with
               waiting(wait):(bound(wait) > bestfound) ?  no;   LLB > GUB.
              );
Loop(i ∈ I+,                                                 Selecting the unfixed location  with
 Loop( j ∈ I- not fxij,                                      the max reduced cost to branch on
              π*_ij = sum(pr,  π+_ip + π-_jp);
              );
Loop(i ∈ I+, loop(j ∈ I- : max π*_ij and not fx_ij,  fx_ij ?  yes  ));

             first2 = 1;                                     Generating two child nodes.
             Loop(fx:first2,
             first2 = 0;
                       fx^0(newnode,i,j) ?  fx^0(current,i,j);
                       fx^1(newnode,i,j)  ?  fx^1(current,i,j);
                       bound(newnode) = LLB;
                       waiting(newnode) ?  yes;
                       fx^0(newnode,fx) ?  yes;

                       newnode(n) ?  newnode(n-1);
                       fx^0(newnode,i,j) ?  fx^0(current,i,j);
                       fx^1(newnode,i,j) ?  fx^1(current,I,j);
                       bound(newnode) = subobj;
                       waiting(newnode) ?  yes;
                       fx^1(newnode,fx) ?  yes;
              );
   );                                                        Ends the current node processing loop
done:(card(waiting) = 0) = 1 );                              Ends the nodes processing loop.
```

Fig. 3. Sketch of the incomplete branch-and price algorithm.

computing upper and lower bounds. Since branch-and-price is an enumeration algorithm enhanced by fathoming based on bound comparisons, the strongest bounds should be the best ones but the mechanism can work with any bound. Nevertheless, the best upper bound might need the resolution of an integer RMP while the best node lower-bound can be obtained by solving the relaxed RMP just

Table 1
Setting options for the algorithm.

Option	
MILP solver (slave problems)	CPLEX 11
MILP solver (optional heuristic GUB computing problems)	Gurobi
Branching rule	On couples ($i \in I^{+}, j \in I^{-}$)
Nodes selection strategy	Best first search
Maximum CPU time per master-slave iteration (s)	60
Multiple columns generated per iteration	Yes
Filtering of columns visiting the same subset of customers	Yes
Time-windows reduction and pre-processing	Yes
Maximum number of master-slave iterations per b&p node	20 (no-root nodes)
Maximum number of branch-and-price inspected nodes	100
Master problem	Partitioning on Eq. (2)
Columns pool	Up to 10000 columns

after the column generation sub-algorithm was unable to produce more profitable columns. So, the best bounds may mean a higher computational cost than weaker bounds. This leads to a trade-off between the CPU time used in computing strong bounds and the size of the explored-tree, that motivates the use of some standard strategies (Desaulniers et al., 2002a,b) to improve the overall algorithmic performance. In this way, to reduce the "tailing-off" effect which consists in a very low convergence-rate at the last iterations of the master-slave recursion, the procedure ends after at most 20 iterations in no-root nodes, thus allowing a larger branching tree.

Every time the column generation procedure ends, the node local lower bound (LLB) is computed and its integrality checked. If the solution is integer and its value is better than the global upper bound (GUB), then the value of the found solution replaces this bound. If not, with the purpose of enhancing the current GUB along the search tree, fast integer solutions are searched and provided by GUROBI. Since we collect several routes per each master-slave iteration by using the *SolnPool* procedure (CPLEX Solver Manual, 2012), we implemented a solutions filter aimed at eliminating sub-optimal solutions delivering the same product orders (i.e. solutions that have the same Z_{ip} values). So, after the resolution of the pricing problem, we feed the RMP with just the best shift delivering a given subset of product orders. This heuristic-nature branch-and-price procedure is focused in efficiency rather than in proving optimality. To turn it on an exact one, limits on the number of master-slave iterations, on the CPU time spent on each slave problem and on the number of inspected nodes must be removed. The node selection strategy is *best-first-search*, that means selecting the node with the lowest LLB from the pool of unsolved subspaces. The algorithm runs in a 2-cores 2.8-Ghz 16-Mbytes RAM PC and the mechanism settings used to solve the problems are summarized in Table 1. To provide an initial solution, feasible routes $i-j-i$ starting from any plant $i \in I^{+}$ and going to any customer $j \in I^{-}$, for any product $p \in P$ are generated. From this initial routes-package the linear RMP can be computed to start the master-slave recursion.

5. Results and discussion

The developed algorithm has been tested by solving several small and medium size instances generated by adapting some examples proposed by Marchetti et al. (2014). Moreover, the procedure was used to solve large instances from a case studio about distribution of fuels on the Santa Fe and Entre Ríos provinces (Argentina).

Marchetti et al. (2014) presented a small example involving 4 customers of liquid oxygen (LOX), 6 customers of liquid nitrogen (LIN), 2 suppliers and a medium size example with 22 customers

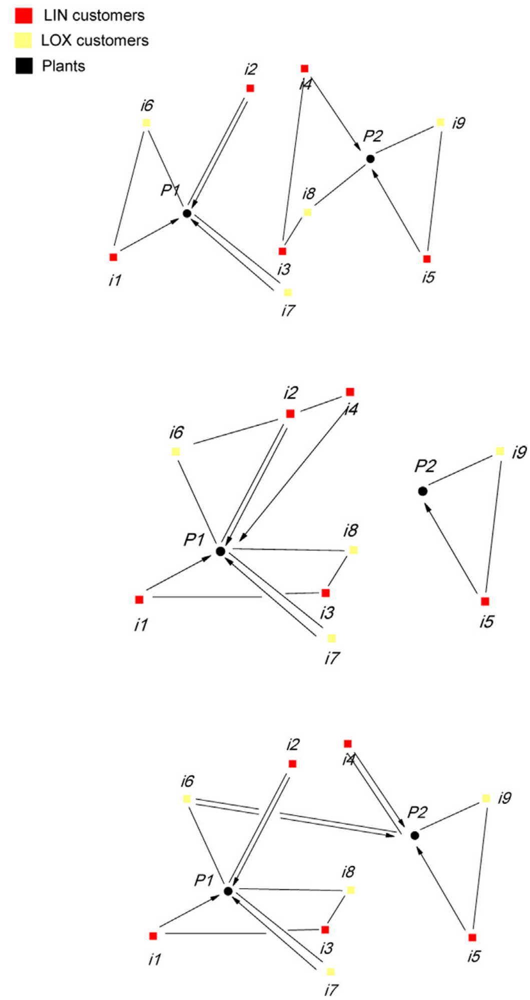

Fig. 4. Solutions to small scale instances.

of LOX, 28 customers of LIN and 3 supply plants in the context of a VMI distribution modality. We first adapted these small examples to the OBR case and then some instances were solved to illustrate how different inventory-levels and demands change the topology of the solutions. The parameters for these instances (plants locations and inventories, customers locations and demands as well as tractor and trailer data) are presented in the tables S1 of the supplementary material section and data about the found solutions are summarized in Table 2. The respective solutions are depicted in Fig. 4. Note that CPU times remain remarkably constants in spite of the strong changes and the rise of cost of routes caused by the tightening of inventory constraints. Note also that the integrality gap did not close in spite that no more routes can be generated.

Afterwards, from the medium-size examples, the following type of instances have been generated to test the above developed algorithm:

1. Realistic instances "R" matching operational conditions presented in Marchetti et al. (2014). From http://dx.doi.org/10.1016/j.compchemeng.2014.06.010 we took the following information: (a) data about plants locations –Table S4-; (b) data about customers locations –Table S6-; data about daily customers-consumption of products –Table S7-. This last table presents information about the daily consumption for 14 different days (t1, t2, ..., t14). With this information and the parameters reported in Tables S2 of the supplementary material section, we generated and solved 21 instances. The first 14 instances corre-

Table 2
Solution data for small scale instances.

	Inventory	levels			Solutions	data	
LIN		LOX		Generated routes	Integer solution	Linear solution	CPU (s)
Plant 1	Plant 2	Plant 1	Plant 2				
∞	∞	∞	∞	62	108.57	104.99	26.4
30	40	20	25	59	118.02	109.03	27.1
50	20	25	20	46	126.18	118.44	27.9

Table 3
Inventory levels and solution data for “R” instances.

LIN inventory			LOX inventory			
P1 50 m^3	P2 25 m^3	P3 50 m^3	P1 25 m^3		P2 12.5 m^3	P3 25 m^3
Instance	Integer solution	Linear solution	Gap (%)	Routes	CPU	Columns
t1	33995	32253	5.1	18	5018.7	703
t2	35386	32373	8.5	20	5906.4	781
t3	35983	33866	5.9	20	5595.6	736
t4	39414	36102	4.4	25	5413.0	654
t5	36128	33895	6.2	21	5062.4	766
t6	37132	34528	7.0	21	4691.3	881
t7	31753	30234	4.8	17	5181.5	837
t8	38561	36359	5.7	23	5647.6	796
t9	34981	33370	4.6	20	4774.9	761
t10	32906	31485	4.3	17	5229.2	713
t11	34821	32396	7.0	20	5139.4	707
t12	34069	32779	3.8	20	6080.2	699
t13	35808	32707	8.7	22	5124.6	1311
t14	35001	32840	6.2	21	6096.5	878
Average	35424	33227	5,9	20	5354,4	801
t1 + t2	51329	48870	4.8	25	1561.4	523
t3 + t4	56989	52802	7.1	28	1442.9	465
t5 + t6	53747	51738	3.7	25	1653.6	504
t7 + t8	52553	50393	4.1	25	1767.9	525
t9 + t10	53728	51850	3.5	25	1696.0	494
t11 + t12	51982	49797	4.2	24	1780.4	534
t13 + t14	54647	51780	5.2	26	950.6	494
Average	53568	51033	4,6	25	1550,4	506

spond to the daily consumptions t1, t2, . . ., t14. The remaining 7 instances are obtained by furnishing the demand of two consecutive days. For example, on the instance t1 + t2, the demand of each customer is obtained by adding the demand of periods t1 and t2. Conversely, the inventory available on plants are obtained by the sum of the inventories for both days. The purpose of these last 7 instances is to evaluate the convenience of delivering a larger shipment to any client in the hope of saving some vehicles with respect to the daily-delivery modality.

2. Pathological instances “P” are generated from “R”-instances but present tight products-inventories in plants P_1 and P_3 located near the geographical area with a large concentration of customer demands. Consequently, the plant P_2 is forced to meet the demand of customers located near the former plants. This pattern should generate larger trips with overlapping trajectories and then these instances should be harder to solve. Data about stocks of plants for these instances are also summarized in Tables S2 of the supplementary material section.
3. Time-window constrained instances “TW” are generated from “R”-instances but consider time windows on some customers. Data about these time windows are presented in Table S2. Since these constraints reduce the solution space of slave problems, these instances should be easier to solve than “R” instances. In turn, these timing constraints leads to longer vehicle tours.

Resolution data for all medium size instances are summarized in Tables 3–5.

From this comprehensive numerical studio, the following conclusions can be reached: (i) All medium size instances showed an integrality gap lower than 10%. They were solved in less than two hours. (ii) The sum of the demands of two consecutive days on a single planning period lead to a sizable saving of used vehicles and therefore of distribution costs in all type of instances. This observation, in turn, lead to the conclusion that is advisable to do a further research about the optimal sizing of deliveries. (iii) Note that not only TW-constrained instances are faster to solve than “R” and “P” instances. Also instances that deliver orders for two consecutive days are solved faster than instances considering just the delivery for a single day. This is, likely, due to the fulfilment of bigger orders that lead to fewer combinations that are assignable on a vehicle.

Large scale instances “L” are generated from a case studio that considers the daily distribution of four fuel types (Premium gasoline, Super gasoline, Eurodiesel and Diesel) to 30 fuel stations (120 orders) in some parts of Santa Fe and Entre Rios provinces (Argentina). Customers to be visited as well as the quantity of fuels to deliver to each one have already been fixed and are problem data. Customers can be supplied from (i) a refinery located in San Lorenzo city or (ii) the fuels distribution-hub located on the Santa Fe harbour (See Fig. 5). Realistic and pathological instances with and without time windows were considered. Data about customers demands and locations as well as data about suppliers stocks and locations are presented in Tables S3.

Multi-compartment trailers with three compartments of 5 m^3 and a tractor of 7.5 m^3 can integrate a vehicle. This introduces and additional complexity respect to the medium scale instances because not only the number of products is doubled but also the number of compartments grows. This implies a larger number of

Table 4
Inventory levels and solution data for “P” instances.

LIN inventory			LOX inventory			
P1 25 m^3	P2 50 m^3	P3 25 m^3	P1 12.5 m^3		P2 25 m^3	P3 12.5 m^3
Instance	Integer solution	Linear solution	Gap (%)	Routes	CPU	Columns
t1	37273	34055	8.6	20	6035.0	785
t2	37583	34338	8.6	21	5051.4	817
t3	38880	33868	5.5	22	5284.0	685
t4	41894	38305	8.6	25	5395.0	662
t5	37190	35500	4.5	19	4654.4	675
t6	39406	36277	7.9	23	5052.2	998
t7	35518	32817	7.6	19	5932.8	713
t8	42549	38606	9.3	24	4655.1	1086
t9	37944	34856	8.1	20	5389.0	759
t10	34015	32122	5.6	19	6075.9	606
t11	35683	33257	6.7	20	5471.2	830
t12	37584	34461	7.8	20	5759.5	647
t13	36829	34036	7.6	21	5392.4	716
t14	37874	34830	8.0	21	5021.4	957
Average	37873	34809	7,5	21	5369,2	781
t1 + t2	56153	54963	2.1	26	1599.0	484
t3 + t4	59624	56486	5.3	28	1185.9	490
t5 + t6	57217	44843	4.1	26	1686.5	499
t7 + t8	55696	54268	2.6	26	1744.8	511
t9 + t10	57942	56087	3.2	26	1735.4	489
t11 + t12	57071	55603	2.6	26	1452.2	535
t13 + t14	57635	55803	3.8	28	812.7	470
Average	57334	54007	3,4	27	1459,5	497

Table 5
Inventory levels and solution data for “TW” scale instances.

LIN inventory			LOX inventory			
P1 50 m^3	P2 25 m^3	P3 50 m^3	P1 25 m^3		P2 12.5 m^3	P3 25 m^3
Instance	Integer solution	Linear solution	Gap (%)	Routes	CPU	Columns
t1	36519	34592	5.3	20	1819.9	580
t2	37251	34354	7.8	22	1414.3	508
t3	36564	34572	5.4	22	1950.6	579
t4	40737	36655	10.0	26	1759.5	533
t5	36314	34223	6.1	20	2047.3	572
t6	39556	37040	6.3	23	2245.8	531
t7	33795	31889	5.6	18	1232.8	539
t8	40102	38342	6.1	25	1872.5	497
t9	37129	35331	4.8	22	1427.8	564
t10	34444	33031	4.1	19	2198.0	487
t11	35714	34105	4.5	20	1534.8	509
t12	34702	34194	1.9	20	1552.2	552
t13	37346	34740	7.2	22	1873.7	494
t14	35725	33916	5.0	20	1529.3	589
Average	36849	34784	5,7	21	1747,0	538,1
t1 + t2	53630	50983	4.9	26	339.8	467
t3 + t4	51370	48599	5.4	24	985.4	532
t5 + t6	54341	52922	2.6	26	623.8	502
t7 + t8	52450	50708	3.3	25	1591.0	513
t9 + t10	54833	53219	2.9	27	497.2	477
t11 + t12	54063	52572	2.7	25	397.0	506
t13 + t14	55532	53800	3.1	27	403.0	455
Average	53745	51829	3,6	26	691,0	493

Table 6
Inventory levels for large scale instances featuring different topologies.

Instance	San Lorenzo inventory levels				Santa Fe inventory levels			
	Premium (m^3)	Super (m^3)	Euro (m^3)	Diesel (m^3)	Premium (m^3)	Super (m^3)	Euro (m^3)	Diesel (m^3)
R	50	120	50	120	25	80	30	80
P	40	100	45	100	30	90	35	90
R-TW	50	120	50	120	25	80	30	80
P-TW	40	100	45	100	30	90	35	90

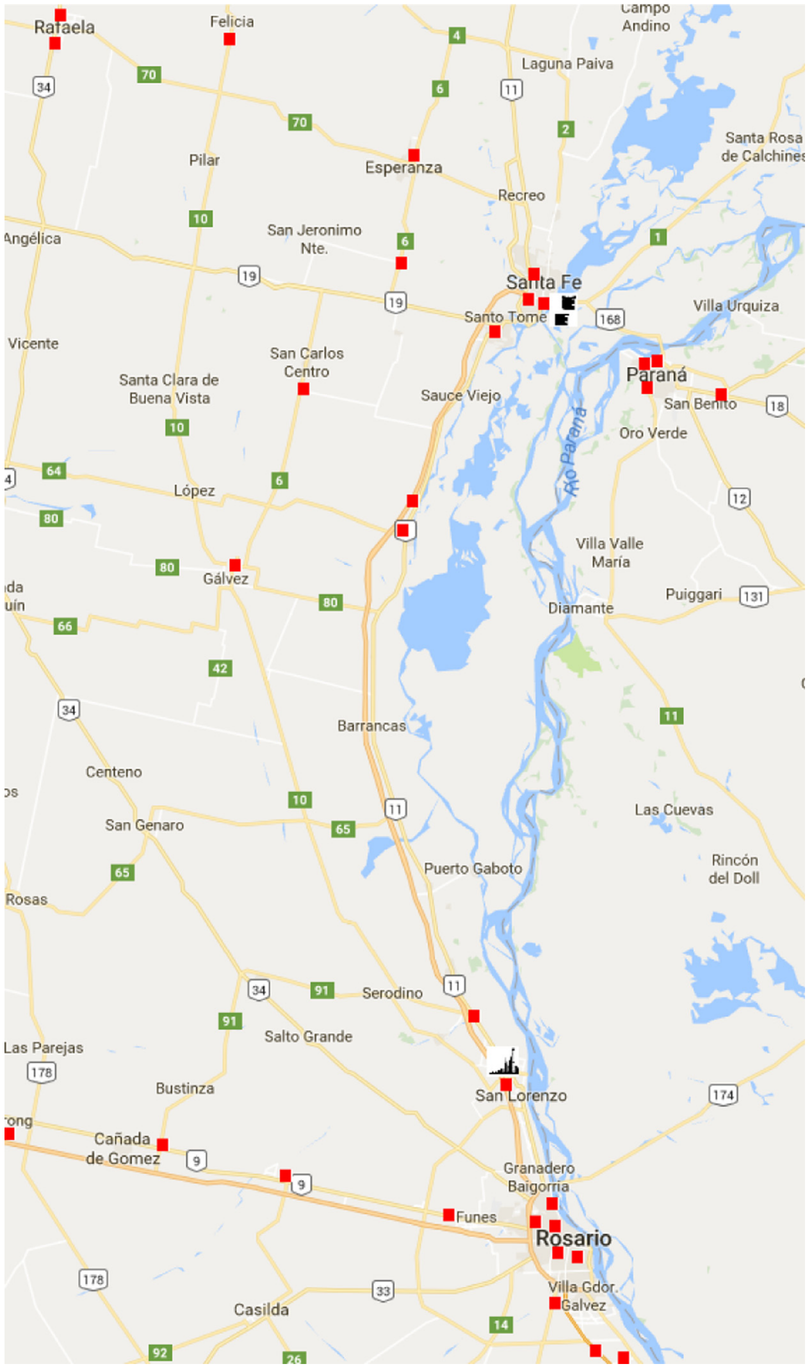

Fig. 5. Geographical view of the distribution and area.

Table 7
Solution data for large scale instances featuring different topologies.

Instance	IS	LS	Gap	Routes	CPU	Columns
R	52200	45951	12.0	27	7856.9	6314
P	53231	44747	16.0	30	10307.7	7033
R-TW	52255	46218	11.5	28	5450.4	5016
P-TW	53805	46019	14.5	30	4935.6	3473

product-to-compartment combination that considerably hardens the resolution of the slave problems. Inventory levels for realistic and pathological instances of large scale with and without time windows are summarized in Table 6 while solution data are shown in Table 7.

It is clear that the gap is enlarged with respect to medium size instances but it can be noted that all solutions are below the 16% threshold. They were obtained in less than three hours even for instances involving more than one hundred orders and vehicles with four compartments.

In order to show the information provided by the solution algorithm, we detailed in Table S4 the solution for the large instance R that involves 27 routes. Two of them use just a tractor and the remaining 25 routes involve vehicles using both a tractor and a trailer. Almost all vehicles are near-full capacity. The solution shows a wide range of trip topologies. We can observe trips to just one or two customers with several products and also we can observe longer trips to several customers for the delivery of a single product. Also, from the perspective of the customers, we observe locations visited by one, by two and even by three vehicles. This observation provides a rational motivation for using optimizing procedures in order to find high quality solutions that assemble shifts with different delivery patterns and almost a full used transport-capacity.

6. Conclusions

This work developed a decomposition strategy based on the paradigm of column generation for planning the distribution of fluid chemicals according to the OBR modality by taking into account inventory constraints on production-plants/distribution-hubs. This one is the first work of a research line aimed at the optimal integration of inventory management and delivery of chemical fluids. The procedure aims at designing the best set of distribution routes for transferring multiple fluid chemicals from plants/hubs to customers on a daily basis. It defines a “vehicle” as tractor-trailer combination and also set the allocation of the products to the compartments of the vehicle. This allows to lower transportation costs through a very efficient use of the compartment of the vehicles. The distribution problem was first modeled as a set partitioning problem with an additional set of inventory constraints and later the model was embedded into an incomplete branch-and-price. The proposed mechanism has been used to solve numerous instances featuring different topologies and sizes. In all examples, the solutions were obtained in reasonable CPU times and show acceptable integrality gaps.

The methodology presented in this work constitutes the building block to use for developing more complex methodologies in order to deal with the optimal management of inventorying and distribution of fluid products. The next research-step is to insert the optimizing tool presented in this work into a framework devised to model and solve the distribution of chemicals according to the VMI modality. In such a modality, the quantities of products delivered to customers are not longer parameters but variables. In addition since this modality involves the planning over a multi-period horizon, the vendor may have the freedom to choose the best day for supplying customers.

Acknowledgements

This work was supported by grants 50120110100315LI from Universidad Nacional del Litoral, grant BID-PICT 2392 from the Agencia Nacional de Promoción Científica y Tecnológica and grant PIP-112201101000940 from Consejo Nacional de Investigaciones Científicas y Técnicas.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at 10.1016/j.compchemeng.2017.06.022.

References

Andersson, H., Hoff, A., Christiansen, M., Hasle, G., Løkketangen, A., 2010. Industrial aspects and literature survey: combined inventory management and routing. Comput. Operat. Res. 37 (9), 1515–1536.

Appelgren, L., 1969. A column generation algorithm for a ship scheduling problem. Transp. Sci. 3 (1), 53–68.

Baita, F., Ukovich, W., Pesenti, R., Favaretto, D., 1998. Dynamic routing and inventory problems. a review. Transp. Res. 32 (8), 585–598.

Barnhart, C., Johnson, E., Nemhauser, G., Savelsbergh, M., Vance, P., 2000. Branch and price: column generation for solving huge integer programs. Oper. Res. 48 (3), 316–329.

Bell, W., Dalberto, L., Fisher, M., Greenfeld, A., Jaikumar, R., Kedia, P., Mack, R., Prutzman, P., 1983. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. Interfaces 13 (6), 4–23.

Benoist, T., Estellon, B., Gardi, F., Jeanjean, A., 2009. High-performance Local Search for Solving Real-life Inventory Routing Problems. Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics., pp. 105–109.

CPLEX Solver Manual. 2012.

Campbell, A., Savelsbergh, M., 2004. A decomposition approach for the inventory routing problem. Transp. Sci. 38 (4), 488–502.

Campbell, A., Clarke, L., Kleywegt, A., Savelsbergh, M., 1998. The Inventory Routing Problem. Fleet Management and Logistics. In: Crainic, T., Laporte, G. (Eds.). Kluwer Academic Publishers, Norwell MA, pp. 95–113.

Campbell, A., Clarke, L., Savelsbergh, M., 2002. Inventory routing in practice. In: Toth, P., Viego, D. (Eds.), The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications, vol. 9. SIAM, Philadelphia, pp. 309–330.

Desaulniers, G., Desrosiers, J., Solomon, M., 2002a. Accelerating strategies in column generation methods for vehicle routing and crew scheduling problems. Essays and surveys in metaheuristics. Operat. Res. Comput. Sci. Interfaces Ser. 15, 309–324.

Desaulniers, G., Desrosiers, J., Solomon, M., 2002b. Accelerating strategies in column generation methods for vehicle routing and crew scheduling problems. essays and surveys in metaheuristics. Operat. Res. Comput. Sci. Interfaces Ser. 15, 309–324.

Desaulniers, G., Desrosiers, J., Solomon, M. (Eds.), 2005. Springer, N. York.

Dondo, R., Mendez, C., 2014. Solving large distribution problems in supply chain networks by a column generation approach. Int. J. Operat. Res. Inf. Syst. 5 (3), 50–80.

Dondo, R., Mendez, C., Cerda, J., 2009. Managing distribution in supply-chain networks. Ind. Eng. Chem. Res. 48 (22), 9961–9978.

Dondo, R., 2012. A new MILP formulation to the shortest path problem with time windows and capacity constraints. Latin Am. Appl. Res. 42, 257–265.

Ghiami, Y., Van Woensel, T., Christiansen, M., Laporte, G., 2015. A combined liquefied natural gas routing and deteriorating inventory management problem. Comput. Logist. 9, 1–104.

Golden, B., Raghavan, S., Wasil, E., 2010. The Vehicle Routing Problem. Latest Advances and New Challenges. Springer, New York.

Grossmann, I., 2012. Advances in mathematical programming models for enterprise-wide optimization. Comput. Chem. Eng. 47, 2–18.

Hemmati, A., Hvattum, L., Christiansen, M., Laporte, G., 2016. An iterative two-phase hybrid matheuristic for a multi-product short sea inventory-routing problem. Eur. J. Oper. Res. 252 (3), 775–788.

Kalvelagen, E., 2009. Some MINLP Solution Algorithms, Downloaded from: http://www.amsterdamoptimization.com/pdf/minlp.pdf.

Kalvelagen, E., 2011. Columns Generation with GAMS, Downloaded from: http://amsterdamoptimization.com/pdf/colgen.pdf.

Lübbecke, M., Desrosiers, J., 2005. Selected topics in column generation. Oper. Res. 53 (6), 1007–1023.

Marchetti, P., Gupta, V., Grossmann, I., Cook, L., Valton, P., Singh, T., 2014. Simultaneous production and distribution of industrial gas supply-chains. Comput. Chem. Eng. 69, 39–58.

Ryan, D., Foster, B., 1981. An integer programming approach to scheduling. In: Wren, A. (Ed.), Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling. , pp. 269–280, North-Holland, Amsterdam.

Salani, M., Vacca, I., 2009. Branch and Price for the Vehicle Routing Problem with Discrete Split Deliveries and Time Windows, Technical. Report TRANSP-OR 091224. Transport and Mobility Laboratory, Ecole Polytechnique Fédérale de Lausanne.

Savelsbergh, M., Sol, M., 1998. Drive: dynamic routing of independent vehicles. Oper. Res. 46 (4), 474–490.

Singh, T., Arbogast, J., Neagu, N., 2015. An incremental approach using local-search heuristic for inventory routing problem in industrial gases. Comput. Chem. Eng. 80, 199–210.

Stålhane, M., Rakke, J., Moe, C., Andersson, H., Christiansen, M., 2012. A construction and improvement heuristic for a liquefied natural gas inventory routing problem. Comput. Ind. Eng. 62 (1), 245–255.

Van der Bruggen, L., Gruson, R., Salomon, M., 1995. Reconsidering the distribution structure of gasoline product for a large oil company. Eur. J. Operat. Res. 81, 460–473.

You, F., Pinto, J., Capón, E., Grossmann, I., Arora, N., Megan, L., 2011. Optimal istribution-inventory planning of industrial gases: I. Fast computational strategies or large-scale problems. Ind. Eng. Chem. Res. 50, 910–2927.

Zamarripa, M., Marchetti, P., Grossmann, I., Singh, T., Lotero, I., Gopalakrishnan, A., Besancon, B., Andreí, J., 2016. Rolling horizon approach for production-distribution coordination of industrial gases supply chains. Ind. Eng. Chem. Res. 55, 2646–2660.